What is claimed is:

1. A system for intrusion detection data collection using a protocol stack multiplexor, comprising:

a hierarchical protocol stack defined within kernel memory space and comprising a plurality of communicatively interfaced protocol layers, each such protocol layer comprising one or more procedures for processing data packets;

a data frame processed through the protocol stack, the data frame comprising a plurality of recursively encapsulated data packets which are each encoded with a protocol recognized by one of the protocol layers; and

a protocol stack multiplexor collecting data directly from the protocol stack from at least one of the processed data packets, comprising:

an interface interfacing directly into at least one such protocol layer through redirected references to the data packet processing procedures comprised within the at least one such protocol layer; and

a logical reference to the processed data packets obtained from the interfaced protocol layer, the logical reference referring to a memory block in the kernel memory space within which the processed data packets are stored and provided to an intrusion detection analyzer executing within user memory space.

2. A system according to Claim 1, further comprising:

a network hardware interface in a link protocol layer logically located at a device end of the protocol stack;

an application software interface in a transport protocol layer logically located at a user end of the protocol stack; and

the protocol stack multiplexor tapping the collected data from the protocol stack between and through the link protocol layer and the transport protocol layer.

3. A system according to Claim 2, wherein the protocol stack comprises a Transmission Control Protocol/Internet Protocol-compliant (TCP/IP) protocol stack.

4. A system according to Claim 1, further comprising:

a read queue associated with each protocol layer storing incoming data frames;

a write queue associated with each protocol layer storing outgoing data frame; and

the protocol stack multiplexor retrieving the logical reference to the processed data packets from at least one of the read queue and the write queue.

5. A system according to Claim 1, further comprising:

a module switch table in the kernel memory space storing the references to the data packet processing procedures comprised within the at least one such protocol layer; and

an initialization module in the protocol stack multiplexor replacing select procedure references in the module switch table with references to data collection procedures in the protocol stack multiplexor.

6. A system according to Claim 5, wherein one such protocol layer comprises a Transmission Control Protocol-compliant (TCP) protocol layer, further comprising:

the initialization module augmenting the procedure references in the module switch table for the procedures for processing data frames for the TCP protocol layer with references to TCP data collection procedures in the protocol stack multiplexor.

7. A system according to Claim 5, wherein one such protocol layer comprises a User Datagram Protocol-compliant (UDP) protocol layer, further comprising:

the initialization module replacing the procedure references in the module switch table for the procedures for processing incoming data frames for the UDP protocol layer with references to UDP data collection procedures in the protocol stack multiplexor.

8. A method for intrusion detection data collection using a protocol stack multiplexor, comprising:

defining a hierarchical protocol stack within kernel memory space and comprising a plurality of communicatively interfaced protocol layers, each such protocol layer comprising one or more procedures for processing data packets;

processing a data frame through the protocol stack, the data frame comprising a plurality of recursively encapsulated data packets which are each encoded with a protocol recognized by one of the protocol layers; and

collecting data directly from the protocol stack from at least one of the processed data packets using a protocol stack multiplexor, comprising:

interfacing directly into at least one such protocol layer through redirected references to the data packet processing procedures comprised within the at least one such protocol layer;

obtaining a logical reference to the processed data packets from the interfaced protocol layer, the logical reference referring to a memory block in the kernel memory space within which the processed data packets are stored; and

providing the logical reference to an intrusion detection analyzer executing within user memory space.

9. A method according to Claim 8, further comprising:

providing a network hardware interface in a link protocol layer logically located at a device end of the protocol stack;

providing an application software interface in a transport protocol layer logically located at a user end of the protocol stack; and

tapping the collected data from the protocol stack between and through the link protocol layer and the transport protocol layer.

10. A method according to Claim 9, wherein the protocol stack comprises a Transmission Control Protocol/Internet Protocol-compliant (TCP/IP) protocol stack.

11. A method according to Claim 8, further comprising:

storing incoming data frames in a read queue associated with each protocol layer;

storing outgoing data frame in a write queue associated with each protocol layer; and

retrieving the logical reference to the processed data packets from at least one of the read queue and the write queue.

12. A method according to Claim 8, further comprising:

storing the references to the data packet processing procedures comprised within the at least one such protocol layer in a module switch table in the kernel memory space; and

replacing select procedure references in the module switch table with references to data collection procedures in the protocol stack multiplexor.

13. A method according to Claim 12, wherein one such protocol layer comprises a Transmission Control Protocol-compliant (TCP) protocol layer, further comprising:

augmenting the procedure references in the module switch table for the procedures for processing data frames for the TCP protocol layer with references to TCP data collection procedures in the protocol stack multiplexor.

14. A method according to Claim 12, wherein one such protocol layer comprises a User Datagram Protocol-compliant (UDP) protocol layer, further comprising:

replacing the procedure references in the module switch table for the procedures for processing incoming data frames for the UDP protocol layer with references to UDP data collection procedures in the protocol stack multiplexor.

15. A storage medium for intrusion detection data collection using a protocol stack multiplexor, comprising:

defining a hierarchical protocol stack within kernel memory space and comprising a plurality of communicatively interfaced protocol layers, each such protocol layer comprising one or more procedures for processing data packets;

processing a data frame through the protocol stack, the data frame comprising a plurality of recursively encapsulated data packets which are each encoded with a protocol recognized by one of the protocol layers; and

collecting data directly from the protocol stack from at least one of the processed data packets using a protocol stack multiplexor, comprising:

interfacing directly into at least one such protocol layer through redirected references to the data packet processing procedures comprised within the at least one such protocol layer;

obtaining a logical reference to the processed data packets from the interfaced protocol layer, the logical reference referring to a memory block in the kernel memory space within which the processed data packets are stored; and

providing the logical reference to an intrusion detection analyzer executing within user memory space.

16. A storage medium according to Claim 15, further comprising:

providing a network hardware interface in a link protocol layer logically located at a device end of the protocol stack;

providing an application software interface in a transport protocol layer logically located at a user end of the protocol stack; and

tapping the collected data from the protocol stack between and through the link protocol layer and the transport protocol layer.

17. A storage medium according to Claim 15, further comprising:

storing incoming data frames in a read queue associated with each protocol layer;

storing outgoing data frame in a write queue associated with each protocol layer; and

retrieving the logical reference to the processed data packets from at least one of the read queue and the write queue.

18. A storage medium according to Claim 15, further comprising:

storing the references to the data packet processing procedures comprised within the at least one such protocol layer in a module switch table in the kernel memory space; and

replacing select procedure references in the module switch table with references to data collection procedures in the protocol stack multiplexor.

19. A storage medium according to Claim 18, wherein one such protocol layer comprises a Transmission Control Protocol-compliant (TCP) protocol layer and a further such protocol layer comprises a User Datagram Protocol-compliant (UDP) protocol layer, further comprising:

augmenting the procedure references in the module switch table for the procedures for processing data frames for the TCP protocol layer with references to TCP data collection procedures in the protocol stack multiplexor; and

replacing the procedure references in the module switch table for the procedures for processing incoming data frames for the UDP protocol layer with references to UDP data collection procedures in the protocol stack multiplexor.

20. A system for detecting network intrusions using a protocol stack multiplexor, comprising:

a network protocol stack comprising a plurality of hierarchically structured protocol layers, each such protocol layer comprising a read queue and a write queue for staging transitory data packets and a set of procedures for processing the transitory data packets in accordance with the associated protocol;

a protocol stack multiplexor interfaced directly to at least one such protocol layer through a set of redirected pointers to the processing procedures of the interfaced protocol layer, further comprising:

a data packet collector referencing at least one of the read queue and the write queue for the associated protocol layer; and

a data packet exchanger communicating a memory reference to each transitory data packet from the referenced at least one of the read queue and the write queue for the associated protocol layer; and

an analysis module receiving the communicated memory reference and performing intrusion detection based thereon.

21. A system according to Claim 20, further comprising:

a module switch table storing a set of pointers to the processing procedures of the interfaced protocol layer; and

an initialization module selectively redirecting the set of pointers to a set of data collection procedures comprised in the protocol stack multiplexor.

22. A system according to Claim 21, further comprising:

a one-way shim redirecting the set of pointers for processing the transitory data packets for one of the read queue and the write queue for the associated protocol layer.

23. A system according to Claim 21, further comprising:

a two-way shim redirecting the set of pointers for processing the transitory data packets for both the read queue and the write queue for the associated protocol layer.

24. A system according to Claim 20, wherein the network protocol stack is a TCP/IP-compliant protocol stack, further comprising:

a set of TCP/IP-compliant protocol layers, selected from the group comprising at least:

a data link protocol layer;

an Internet (IP) protocol layer;

an Transmission Control Protocol (TCP) layer; and

a User Datagram Protocol (UDP) layer.

25. A method for detecting network intrusions using a protocol stack multiplexor, comprising:

executing a network protocol stack comprising a plurality of hierarchically structured protocol layers, each such protocol layer comprising a read queue and a write queue for staging transitory data packets and a set of procedures for processing the transitory data packets in accordance with the associated protocol;

interfacing a protocol stack multiplexor directly to at least one such protocol layer through a set of redirected pointers to the processing procedures of the interfaced protocol layer, further comprising:

referencing at least one of the read queue and the write queue for the associated protocol layer; and

communicating a memory reference to each transitory data packet from the referenced at least one of the read queue and the write queue for the associated protocol layer; and

receiving the communicated memory reference into an analysis module and performing intrusion detection based thereon.

26. A method according to Claim 25, further comprising:

storing a set of pointers to the processing procedures of the interfaced protocol layer into a module switch table; and

selectively redirecting the set of pointers to a set of data collection procedures comprised in the protocol stack multiplexor.

27. A method according to Claim 26, further comprising:

redirecting the set of pointers for processing the transitory data packets for one of the read queue and the write queue for the associated protocol layer.

28. A method according to Claim 26, further comprising:

redirecting the set of pointers for processing the transitory data packets for both the read queue and the write queue for the associated protocol layer.

29. A method according to Claim 25, wherein the network protocol stack is a TCP/IP-compliant protocol stack, further comprising:

defining a set of TCP/IP-compliant protocol layers, selected from the group comprising at least:

a data link protocol layer;

an Internet (IP) protocol layer;

an Transmission Control Protocol (TCP) layer; and

a User Datagram Protocol (UDP) layer.